U.P.B. Sci. Bull., Series C, Vol. 83, Iss. 2, 2021 ISSN 2286 – 3540

HYBRID GLOBAL ILLUMINATION: A NOVEL APPROACH COMBINING SCREEN AND LIGHT SPACE INFORMATION

Cristian LAMBRU[1], Anca MORAR[1], Florica MOLDOVEANU[1], Victor ASAVEI[1], Silviu IVAȘCU[1]

Modern real-time computer graphics, augmented reality enhanced devices or virtual reality head mounted displays expect a high degree of photo-realism for an immersive experience. One of the most important components of the photo-realistic effect is represented by the simulation of the light transport, which for real-time applications is widely known as global illumination. In this paper we present a novel hybrid global illumination method that simulates the light transport using information available both in screen space and light space. We define a strategy to combine both sources of information to obtain both the diffuse and specular components of the indirect light, together with the lighting attenuation produced by ambient occlusion. We present different extensions to accommodate multiple types of light, to produce subsurface scattering and the direct lighting of area lights. Also, we included different optimization strategies to scale the approach for multiple lights. We evaluate our method based on both qualitative and quantitative criteria and analyze the results.

Keywords: global illumination, reflective shadow mapping, screen space directional occlusion, screen space reflections, ambient occlusion

1. Introduction

The effect of photo-realism is very important in graphic applications and with the advance of graphics processing units (GPUs), this effect is suitable for real-time graphic applications. The photo-realistic effect requires several components, but one of the most important is the indirect lighting. This is the lighting contribution that doesn't come directly from a light source, but comes after several bounces on the geometry. The effect of photo-realism in real-time applications is widely known as global illumination.

This process requires information for the entire geometry scene and complex geometric operations to simulate the light transport using different approximations. Over time, several classes of techniques that use different approximations for the scene geometry and different gathering strategies to acquire the indirect lighting were developed.

[1] Faculty of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: andrei.lambru@upb.ro, anca.morar@cs.pub.ro, florica.moldoveanu@cs.pub.ro, victor.asavei@cs.pub.ro, ivascu.silviu10@gmail.com

We present the related work in the global illumination field in Section 2. In Section 3 we present a new hybrid approach that computes the global illumination using two framebuffers obtained from rendering the scene from two different positions: the light and the observer position. We propose a strategy to sample those framebuffers in order to acquire both the diffuse and specular components of the light and the ambient occlusion map to attenuate the indirect light. In Section 4, we propose several extensions of this approach to accommodate multiple light types, to produce subsurface scattering and the direct lighting of area lights. Also, we propose several optimizations to scale the approach for multiple lights. We evaluate this approach and present the results in Section 5 and our conclusions in Section 6.

2. Related work

The field of global illumination contains several techniques that suit different purposes. Appel [1] proposed a ray-tracing technique to acquire the indirect lighting in the reflected and transmitted directions. Unfortunately, this approach provides very good results in glossy environments, but lacks the ability to produce the diffuse component of the indirect light. Kajiya [2] extended this approach with a set of randomly traced rays to acquire the indirect light from multiple zones of the scene. Other approaches discretize the scene geometry into patches and evaluate the indirect light influence for each possible pair of patches. Cohen and Wallace [3] present a comprehensive analysis of this class of techniques, known as Radiosity. Another class of techniques use the ray-tracing technique to spread a set of photons into the geometry and use a statistical approach [4] or a gathering approach [5] to compute the indirect lighting with the information acquired with the photon set.

All the techniques described above offer photo-realistic results, but are not suitable for real-time application having too much complexity for current GPUs. However, several data structures [6-8] were proposed to cache intermediary results provided by those techniques and to use them in real-time. A true real-time solution was proposed by Dachsbacher and Stamminger [9]. They used a separate pass to render the scene from the position of the light with multiple information per pixel. The resulted framebuffer is known as Reflective Shadow Map (RSM). This technique produces a set of lights, represented by the entire resolution of the map, where each pixel becomes a Virtual Point Light (VPL). In the second pass where the scene is rendered from the observer position, each object is shaded with a sampling set of VPLs acquired from the RSM. Multiple sampling strategies [10,11] were proposed over time. Moreover, the RSM was used only as a source of VPLs in different approaches [12,13].

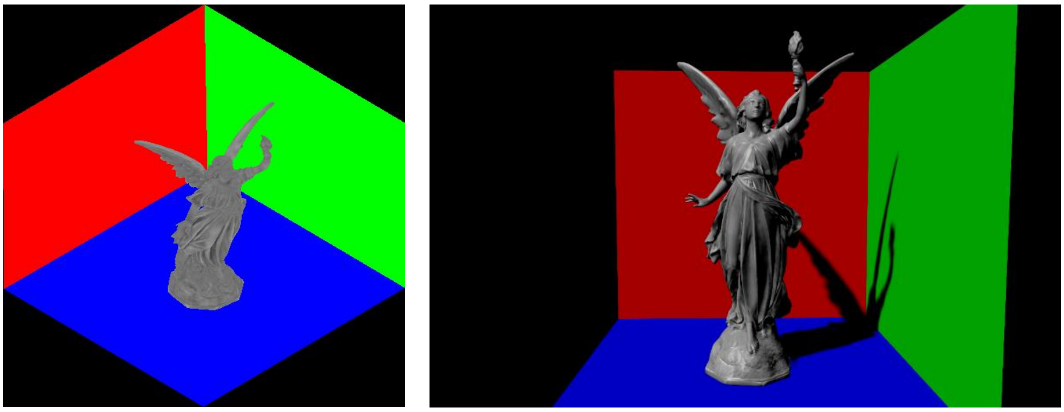
Another class of real-time global illumination techniques uses the voxel representation of the scene geometry. This data structure is very flexible and could provide a coarse approximation of the geometry. Thiedemann et al. [14] proposed a ray-tracing process to acquire indirect lighting information from the voxel volume. Crassin et al. [15] proposed a cone-tracing approach that uses multiple levels of detail for the voxel volume and takes advantage of the hardware accelerated trilinear filtering provided by current GPUs for smooth indirect lighting.

The screen space information can be a faithful representation of the scene geometry. It is always computed and provides detailed geometry, without consuming considerable resources. However, it is not an exact representation of the geometry and is limited by the information visible from the observer position. Several techniques [16-18] that use the screen space information for indirect lighting computation were designed. Ritschel et al. [19] proposed a direct sampling approach, called Screen Space Directional Occlusion (SSDO) to acquire both the direct and the indirect lighting. However, the sampling strategy represents a coarse approximation of the path-tracing approach and could only provide the indirect diffuse component. To obtain the indirect specular component, Sousa et al. [20] first proposed to use a 2D ray-tracing approach directly in screen space. This approach was developed at the same time by multiple companies. The class of techniques that use the 2D ray-tracing approach to compute the specular component of the indirect light is known as the Screen Space Reflections (SSR).

2.1 Reflective Shadow Mapping

Dachsbacher and Stamminger [9] proposed to render the scene from the position of the light in the lighting direction. The projection used to render the scene is chosen according to the light type, a perspective projection for a spot light and an orthographic one for a directional light. The resulted map, called Reflective Shadow Map (RSM), has multiple information per pixel: position in world space, normal vector in world space, the reflected flux and the depth. The reflected flux represents the albedo of the fragment according to the angle between the direction of the incoming light and the direction along which the light source has the highest intensity. Those directions are the same for a directional light and the reflected flux represents the albedo of the fragment (Fig. 1a).

The authors proposed a sampling strategy to acquire the indirect diffuse component of the light with the RSM. They used a sampling pattern in a local 2D space with coordinates in the interval $[-1,1]$. When the scene is rendered from the observer position, the position of each shaded fragment is projected into the light space to obtain the corresponding 2D position on the RSM. This position is used as the center of the sampling pattern. Each sample is transformed from its local space into RSM space to obtain the corresponding pixel from the RSM. Each pixel becomes a VPL and is evaluated for the shaded fragment, according to the positions, normal vectors and albedos of the fragment and the VPL.



a) Reflected flux map of RSM b) Direct light map

Fig. 1. Stanford Lucy model in a 3-sided Cornell box with three different colors. The scene is rendered from the light position (a) and from the observer position (b). The scene is lighted by a single directional light.

The main advantage of this method is the time performance in case of a small number of lights, due to the requirement of only one additional rendering of the scene for each light. Another advantage is its correlation with the shadow mapping technique. However, it comes with several drawbacks. First, it acquires a single bounce for the indirect light. Moreover, it produces visual artifacts caused by the fixed 2D sampling pattern.

2.2 Screen Space Directional Occlusion

Ritschel et al. [19] proposed to use the information of the screen space because it is already rendered and provides detailed geometry. However, the process known as deferred rendering [21] is required to cache multiple information per pixel: position in world space, normal vector in world space, albedo and depth.

They proposed to use a similar sampling approach like the one presented in the previous sub-section. They used a sampling pattern in a local 3D space. Each sample resides inside a hemisphere with radius 1 in the direction of the local Y axis. When the scene is rendered from the observer position, the pattern is transformed from its local space into screen space. Each sample is transformed from the local space into the object local space, where the position of the fragment corresponds to the origin of the sample local space and the fragment normal corresponds to the sample local Y axis. Each pattern is transformed into world space and finally into screen space to obtain the corresponding pixel from the sample position. Each pixel becomes a VPL and is evaluated similarly to the previous sub-section.

The source of information is the direct light map from the observer position (Fig. 1b), obtained at a preliminary pass. The acquisition of the indirect lighting with the direct light map simulates a second bounce of light and is required for a physically correct approximation. The main advantage of this method is the use of the already computed direct light buffer as a source of information. Unfortunately, this approach is limited by the geometry visible from the observer position and the geometry that is not present on the screen cannot contribute to the indirect specular light.

2.3 Screen Space Reflections

All the techniques from the Screen Space Reflections compute the specular component of the indirect light using the screen space information. To acquire the indirect specular light for a point x, Appel [1] proposed to use a ray in world space along the reflected direction according to the normal on point x. The ray is intersected with the scene geometry to provide the information in the reflected direction.

Unfortunately, this process has a negative impact on performance and Sousa et al. [20] proposed to acquire this information using the already rendered framebuffer that was obtained at a preliminary pass. This approach requires the use of deferred rendering [21] to cache multiple information per pixel: position, normal vector, albedo and depth. They computed the reflected direction and projected this direction in screen space. The geometry intersection for the resulted 2D direction is checked using the pixel information and the sampling positions along the reflected direction in world space. This technique comes with the same advantages and disadvantages as the Screed Space Directional Occlusion.

2.4 Ambient Occlusion

The occlusion of the indirect light is necessary to avoid over-illuminating the scene. This process can be approximated using a localized approach that reduces the intensity of the indirect lighting on a point x according to the amount of geometry in the upper hemisphere of that point. A point without close geometry will receive the maximum intensity of the indirect light and a point occluded by geometry in half of the hemisphere (e.g. corners) will receive half the intensity.

Miller [22] first proposed this approach to approximate the occlusion of the indirect light and over time several techniques were proposed. A comprehensive analysis was described by Méndez-Feliu and Sbert [23]. Mittring [24] first proposed to use the screen space information to acquire the ambient occlusion map. He used a sample pattern in a local 3D space that was projected in screen space to obtain framebuffer pixels. Those pixels are tested against the sampling positions in world space to obtain the number of samples that reside behind the visible geometry in screen space.

3. Method

The direct lighting is computed with the illumination model proposed by Phong [25]. We kept this model for the indirect lighting as well and separated it into diffuse and specular components. Furthermore, we used the ambient occlusion to approximate the occlusion of the indirect lighting according to the surrounding geometry. The computation of the indirect light can be described using:

$$\boldsymbol{L_{ind}}(\boldsymbol{x}) = \left(\boldsymbol{\alpha_x L_{dif}}(\boldsymbol{x}) + \boldsymbol{\sigma_x L_{spec}}(\boldsymbol{x})\right)\boldsymbol{O}(\boldsymbol{x}) \quad (1)$$

where $\boldsymbol{x}$ is the position of the evaluated point, $\boldsymbol{L_{ind}}(\boldsymbol{x})$ is the indirect lighting, $\boldsymbol{\alpha_x}$ is the albedo at position $\boldsymbol{x}$, $\boldsymbol{\sigma_x}$ is the specular coefficient at position $\boldsymbol{x}$, $\boldsymbol{L_{dif}}(\boldsymbol{x})$ and $\boldsymbol{L_{spec}}(\boldsymbol{x})$ are the diffuse and specular components of the indirect light at position $\boldsymbol{x}$ and $\boldsymbol{O}(\boldsymbol{x})$ is the ambient occlusion attenuation value at position $\boldsymbol{x}$.

The $\boldsymbol{\alpha_x}$ and $\boldsymbol{\sigma_x}$ terms are material specific properties at position $\boldsymbol{x}$ and represent local information. The computation of the indirect lighting and occlusion terms will be detailed in the following sub-sections.

3.1 Indirect diffuse lighting

To acquire the indirect diffuse lighting, we apply both the Reflective Shadow Mapping and Screen Space Directional Occlusion techniques. To avoid the accumulation of the same VPLs multiple times for the same fragment, we use different sample patterns which do not overlap. However, for such an approach it is required that both sample patterns to be in the same local space. We chose the local 3D space used by the SSDO technique.

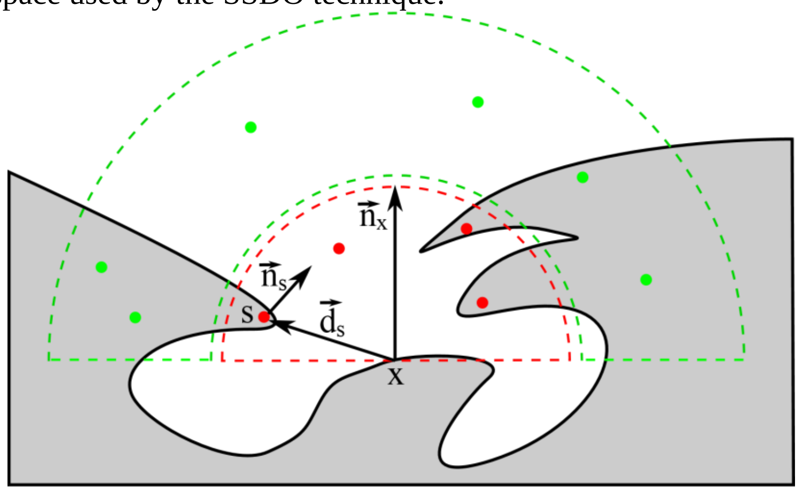
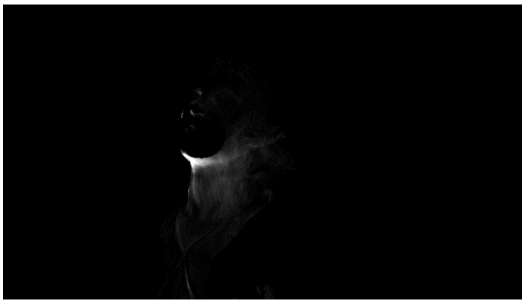


Fig. 2. The indirect diffuse light sampling strategy that combines the RSM and SSDO approach. The SSDO samples are scattered inside the small red hemisphere and the RSM samples are scattered inside the larger green hemisphere.

a) Indirect diffuse lighting acquired using the RSM (200 samples)



b) Indirect diffuse lighting acquired using the screen space information (200 samples)

c) Indirect diffuse lighting acquired using both the RSM and the screen space

d) Direct and indirect diffuse lighting

Fig. 3. The indirect diffuse lighting acquired with our sampling strategy (c). This approach uses a weighted additive blend between the indirect diffuse lighting results produced by the Reflective Shadow Mapping (a) and Screen Space Directional Occlusion (b) techniques.

Also, to ensure that the two sample patterns do not overlap, we used two different zones of sampling that do not intersect. We compute a small hemisphere for the SSDO pattern and a larger hemisphere for the RSM, without the zone already used for the SSDO. This strategy is described in Fig. 2. The reason behind this decision is to benefit from the superior details offered by the screen space. Its resolution is higher than the RSM and provides more geometry details. Moreover, there is a performance problem when a large texture is randomly sampled. The caching mechanism of the GPU's Texture Mapping Unit (TMU) is overloaded when the sampling positions are uniformly distributed on texture space and has a huge negative impact on performance. However, when the samples are closer in texture space, the caching mechanism is efficient. Our small hemisphere ensures that this happens for most scenarios. The RSM resolution is smaller and this problem doesn't have the same negative impact on performance, therefore a larger hemisphere was chosen for it. The smaller resolution is more suitable for large distances due to the fact that the pixel provides more information for that zone. The results of this approach, illustrated for each technique, are presented in Fig. 3.

Unfortunately, this approach implies that the sampling strategy used by the SSDO needs to be adapted for the RSM and it affects the visual results. The original proposal tries to ensure that different pixels from the RSM are accumulated from an acceptably large neighborhood. It is possible for a pixel to be accumulated multiple times, but the 2D local space of the sample pattern makes these overlaps rare. Unfortunately, the sampling strategy of the SSDO could not ensure such restriction for the RSM and according to the perspectives used, there can be many overlaps. However, this error is acceptable for our purpose. A higher resolution of the RSM helps, but it is not required due to the fact that supplementary pixels from a higher resolution would have similar positions, normal vectors and albedos like the ones from a smaller resolution.

The diffuse component of the indirect lighting is computed using:

$$L_{dif}(x) = \frac{L_{rsm}(x)N_{rsm} + L_{ssdo}(x)N_{ssdo}}{N_{rsm} + N_{ssdo}} \quad (2)$$

where $L_{rsm}(x)$ is the indirect diffuse lighting acquired with the Reflective Shadow Mapping technique, $L_{ssdo}(x)$ is the indirect diffuse lighting acquired with the Screen Space Directional Occlusion technique and N_{rsm}, N_{ssdo} represent the numbers of samples used for the techniques.

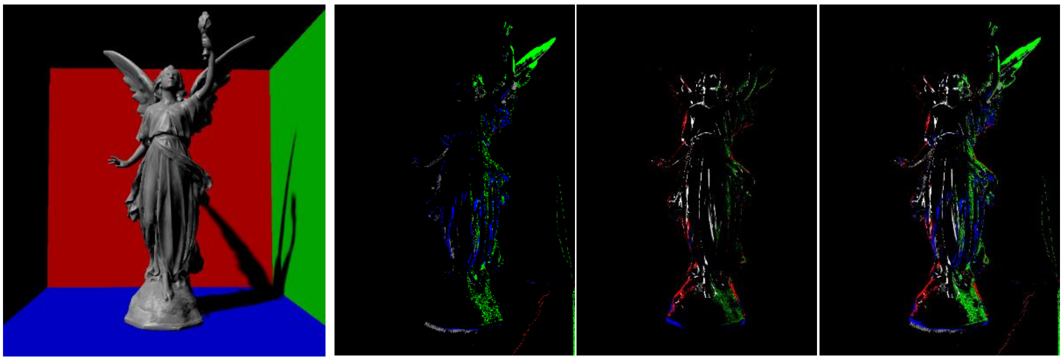
The computation of the indirect lighting using the sample set, regardless of the data structure on which the samples are projected, is done using a similar approach like the one proposed by Dachsbacher and Stamminger [9], adapted for a local 3D space:

$$L(x) = \frac{1}{N}\sum_{i=1}^{N} \alpha_{s_i} \frac{\langle \vec{d}_{s_i} \bullet \vec{n}_x \rangle \langle -\vec{d}_{s_i} \bullet \vec{n}_{s_i} \rangle}{||x - s_i||^2} \quad (3)$$

where N is the number of samples, s_i is the sample position, α_{s_i} is the albedo of the sample, $\vec{n}_x$, $\vec{n}_{s_i}$ are the normal vectors at position x and s_i and $\vec{d}_{s_i}$ is the vector direction between x and s_i, presented in Fig. 2. $\vec{u} \bullet \vec{v}$ represents the dot product of vectors $\vec{u}$ and $\vec{v}$, $\langle a \rangle$ represents the value of a clamped to the minimum value of 0 and $||\vec{u}||$ represents the length of vector $\vec{u}$.

3.2 Indirect specular lighting

We use the approach proposed by Sousa et al. [20] to acquire the specular component of the indirect light in screen space and the specular component from the RSM. Both data structures provide the same information. The 2D ray-tracing process starts in the same direction for both data structures and both of them provide



a) Direct lighting b) RSM c) SSR d) Hybrid

Fig. 4. The indirect specular lighting acquired with our hybrid approach (d), which uses an exclusive blending between the indirect specular lighting results acquired with a 2D ray-tracing process on RSM (b) and screen space (c). The latter uses the direct lighting map (a) as source.

the same intersecting geometry when it is present on the framebuffer. We apply an exclusive blending approach that combines both of these results. We acquire the indirect specular light from the screen space first and, only if the ray-tracing process does not intersect geometry, we acquire the indirect specular light from the RSM. The reasons behind this strategy are similar to those for the indirect diffuse lighting. The screen space has a higher resolution than the RSM and contains detailed geometry. It provides sharper results compared with the same approach applied on the RSM. The indirect specular light simulation requires a high level of precision, therefore the screen space is the most suitable framebuffer for this process. When the intersecting geometry is not present in screen space, a new attempt is made to acquire a coarser version of the indirect specular light from the RSM. However, it is possible that there is no intersecting geometry or this geometry is not present on any of the framebuffers, therefore there will be no indirect specular light for the evaluated fragment. The results of this strategy, illustrated for each technique, are presented in Fig. 4.

The specular component of the indirect light is computed as follows:

$$L_{spec}(x) = L_{ssr}(x) + bL'_{rsm}(x) \tag{4}$$

where $L_{ssr}(x)$ is the indirect specular lighting acquired with the Screen Space Reflections technique, $L'_{rsm}(x)$ is the indirect specular lighting acquired using the same 2D ray-tracing process applied on RSM and b is the exclusive term which is 1 only when the $L_{ssr}(x)$ term is 0 and 0 otherwise.

a) RSM b) Screen space c) Hybrid

Fig. 5. The hybrid ambient occlusion map (c) acquired using the blending of two other ambient occlusion maps obtained from RSM (a) and screen space (b). The latter two were acquired with the sampling strategy proposed by Mittring [24].

3.3 Ambient occlusion

We use the technique proposed by Mittring [24] to compute the ambient occlusion map for both the screen space and RSM as the source of geometry for the sample pattern evaluation. This process is applied for all fragments of the framebuffer. We use the weighted average proposed by Vardis et al. [26] to combine the ambient occlusion maps acquired with the two data structures. This weighted average was specially designed to combine multiple ambient occlusion maps acquired from different positions. The resulted ambient occlusion map is enhanced with a blurring post-processing effect to soften the noise resulted from the sampling approach. The visual results for each technique are presented in Fig.5.

4. Extensions

4.1 Different light types

Our approach can be adapted to accommodate the acquisition of the indirect lighting for different light types. Two such light types are the directional and spot light type. The methodology described in section 3 is mostly the same for both types. The only significant difference between them are the acquisition of the RSM and the light space used to project the sample pattern on RSM. The RSM acquisition requires an orthographic projection for the directional light, respectively a perspective projection for the spot light. The SSDO and SSR techniques are the same for both light types.

Another light type is the point light, which is described by a position around which light is scattered in all directions. To accommodate this light type with our approach, several RSMs are acquired. The scene is rendered several times from the position of the light along different directions to cover the entire sphere in which the light is scattered. This is required to acquire all the lighted areas around the light position. However, according to the projection itself, the number of renderings could differ. Apart from the use of several RSMs, each with its own projection space, the SSDO and SSR techniques remain the same as described in Section 3.

4.2 Multiple lights

Our approach can be extended to acquire the indirect lighting for multiple lights of any type. However, to optimize this process, we included several optimizations to avoid the computing of the same information multiple times.

The first optimization computes the indirect diffuse lighting with the SSDO technique only once. It is required to compute the RSM for each light, therefore the indirect diffuse lighting from the RSM is computed for each light. However, the SSDO is independent of the light count and requires only the direct light map. Since each light can have a contribution to the direct light, this map is created before the lighting pass starts and is updated for each light. After each light is processed, the SSDO sampling is done for the final direct light map which accumulates the direct lighting from each light. The $L_{rsm}(x)$ term in Eq. 2 becomes:

$$L_{rsm}(x) = \sum_{i=1}^{lights} \frac{L_{rsm_i}(x) N_{rsm_i}}{N_{rsm}} \tag{5}$$

where $L_{rsm_i}(x)$ is the indirect diffuse lighting acquired for the i-th RSM, N_{rsm_i} is the number of samples used for the i-th RSM and N_{rsm} is the number of samples for all RSMs.

The second optimization is applying the 2D ray-tracing process in screen space only once. The 2D ray-tracing process on RSM is required for each light because the RSM itself is generated for each light. Unfortunately, due to the fact that the RSM ray-tracing process is done only if the screen space ray-tracing failed to provide any intersection, the latter process is required as a preliminary test for each light. However, the process itself is the same for each light. To optimize this process, we computed a 2D ray-tracing intersection map for the geometry that is visible in screen space, before the lighting pass starts. When a light is processed, we check this intersection map and the intermediary direct light map to find if there is any information in screen space. Only if there is no intersection or no lighted intersection we proceeded to compute the indirect specular light on the RSM.

The third optimization is applying the sampling strategy proposed by Mittring [24] only once for screen space. The ambient occlusion map acquired with the RSM is different for each light. However, the ambient occlusion map for screen space needs to be computed only once. Moreover, the weighted average proposed by Vardis et al. [26] is additive. For this reason, we accumulate both the weighted ambient occlusion value and the weight value itself on a two-color component ambient occlusion map for all the lights. After the lighting pass is finished, we compute the ambient occlusion map for screen space and combine the latter results with those accumulated for the lights.

4.3 Subsurface scattering

The process of rendering translucent materials requires supplementary information for the scene geometry. Dachsbacher and Stamminger [27] introduced Translucent Shadow Maps (TSMs), which store the transmitted light inside the geometry. They proposed a sampling strategy to acquire the subsurface scattering of the direct light. There are several techniques [28] that take advantage of a set of Gaussian filters to approximate the subsurface scattering in screen space. However, this process is not physically correct and acquires only the subsurface scattering of the direct light, without the possibility to acquire information from the geometry behind the object. Nalbach et al. [29] introduced the concept of deep screen space, which represents a set of surfels for the geometry visible from the observer position. They proved that this approach can provide plausible results for the subsurface scattering. To avoid the introduction of such approaches, which require supplementary effects, we chose to acquire the subsurface scattering of the indirect light with a 2D ray-tracing process on the RSM. We compute the transmitted direction of the light for the shaded fragment and use the ray-tracing process on the RSM to acquire the intersected geometry. It is possible to use the same approach to check the screen space, but the intersected geometry is usually behind the visible objects, therefore we only search the intersection on the RSM.

4.4 Area lights

The area light spreads light from the entire surface described by its geometry. This light type contains geometry and emits light by itself. The purpose of a global illumination technique is to acquire the direct lighting of the area light, which requires a process similar to that of acquiring indirect light for the geometry that does not emit light.

Unfortunately, because this light type doesn’t have a position that acts as the source of the light, it cannot have a corresponding projection to acquire all the lighted areas. Therefore, the RSM sampling strategy could not be used to acquire the direct lighting, because it is not possible to acquire the RSM itself. However, the SSDO approach can acquire the direct lighting of an area light. The only requirement is to take into account the emissive component when the direct light map is computed. This approach acquires the direct lighting for the geometry of the area lights that resides in screen space. Unfortunately, the approach is limited by the geometry visible from the observer position and any geometry from an area light that is not present on the screen doesn't contribute with its direct light.

5. Results

We implemented the described approach using the OpenGL 4.6 graphics API and the C++ programming language. We evaluated the implementation using the scene presented in Fig. 1, that contains 33,452 triangles, on an Nvidia GTX 1070 GPU with two different resolutions: $\mathbf{1280 \times 720}$ and $\mathbf{1920 \times 1080}$. We used only a single directional light.

The Reflective Shadow Mapping and Screen Space Directional Occlusion techniques were enhanced with the screen space interpolation optimization proposed by Dachsbacher and Stamminger [9]. We used a radius of 3 for the screen sampling and a radius of 30 for the RSM sampling. The results are presented in Table 1.

Table 1

The frame rate for different rendering framebuffer resolutions with various sample pattern sizes (R is the number of samples for RSM and S is the number of samples for SSDO)

		S=50	S=100	S=150	S=200
1280x720	R=50	125 fps	105 fps	90 fps	79 fps
	R=100	115 fps	98 fps	85 fps	75 fps
	R=150	107 fps	92 fps	81 fps	71 fps
	R=200	100 fps	86 fps	76 fps	68 fps
1920x1080	R=50	54.3 fps	44.5 fps	37.5 fps	32.7 fps
	R=100	50.4 fps	42.9 fps	35.9 fps	31.3 fps
	R=150	47.5 fps	39.7 fps	34.1 fps	30.0 fps
	R=200	44.5 fps	37.7 fps	32.7 fps	28.9 fps

It is visible that a larger sample pattern for both techniques has a negative impact on performance for obvious reasons. It is also visible that a higher resolution has a huge negative impact on performance. However, the reason behind this is a little complicated. Table 2 presents the rendering times for each indirect diffuse lighting techniques. The actual impact on performance is up to 2x, but the SSDO technique has higher rendering times and 70% of the rendering time for the indirect diffuse lighting is consumed by it.

We evaluated the indirect specular lighting computations. Table 3 presents the rendering times for the generation of the SSR intersections map, presented in Section 4.2 and the actual SSR pass which combines the indirect specular lighting from both the RSM and the screen space. It is visible that the framebuffer resolution has a negative impact on performance. However, the resolution has a smaller impact for the 2D ray-tracing process on RSM compared with the screen space approach.

Table 2

The rendering times (ms) for each indirect diffuse lighting technique for different rendering framebuffer resolutions with various sample pattern sizes

		RSM	SSDO
1280x720	50 samples	0.82 ms	1.57 ms
	100 samples	1.52 ms	2.98 ms
	150 samples	2.18 ms	4.29 ms
	200 samples	2.84 ms	5.72 ms
1920x1080	50 samples	1.30 ms	3.70 ms
	100 samples	2.39 ms	5.86 ms
	150 samples	3.45 ms	8.80 ms
	200 samples	4.50 ms	11.21 ms

Table 3

The rendering times (ms) for the indirect specular lighting with different rendering framebuffer resolutions

	SSR intersections map generation	SSR (RSM+screen)
1280x720	1.03 ms	1.28 ms
1920x1080	2.69 ms	2.23 ms

We evaluated the ambient occlusion map computations (Table 4). The ambient occlusion approach was evaluated using 32 samples for both the RSM and screen space. The radius for both sample patterns was 1. It is visible that a higher framebuffer resolution has a negative impact on performance. Moreover, the impact is bigger for the screen space approach due to the TMU overloading. However, the overall impact is significantly smaller compared to the computation of the indirect lighting components due to the use of a small radius for the sampling pattern.

Table 4

The rendering times (ms) for the ambient occlusion techniques with different rendering framebuffer resolutions

	RSM AO	Screen AO
1280x720	0.43 ms	0.46 ms
1920x1080	0.58 ms	0.79 ms

We also evaluated the original RSM and SSDO techniques (Table 5). Both techniques were evaluated with the ambient occlusion approach proposed by Mittring [24] applied on the RSM and screen space respectively. The original RSM was implemented using a local 2D space for the sample pattern and the original SSDO technique was evaluated with two different radii, one to simulate the same radius as in our approach for the SSDO alone and the other to simulate both the

RSM and SSDO sampling processes. As can be observed, the impact of the radius is minor for the original SSDO. Unfortunately, a higher resolution has up to a 3x impact on performance for both original techniques. The impact of the framebuffer resolution for the original SSDO is significantly higher compared to the impact on the original RSM.

It is also visible that both original RSM and SSDO techniques have better performances compared to our hybrid approach. However, our approach provides better visual results. Fig. 6 presents those results for the techniques along with two versions of the SSDO technique with two different radii of the sample pattern as described in Table 5.

Table 5

The frame rate for the original RSM and SSDO techniques for different rendering framebuffer resolutions with various sample pattern sizes

		Original RSM	Original SSDO (r=3)	Original SSDO (r=30)
1280x720	50	269 fps	182 fps	179 fps
	100	237 fps	141 fps	135 fps
	150	213 fps	116 fps	109 fps
	200	195 fps	98 fps	92 fps
1920x1080	50	145 fps	72.5 fps	71.4 fps
	100	129 fps	56.1 fps	54.9 fps
	150	116 fps	45.9 fps	44.3 fps
	200	106 fps	38.6 fps	37.3 fps

It is visible that the SSDO technique could not provide physically correct results because it relies entirely on information that is present in screen space. This problem is visible for the Stanford Lucy right wing where the green color bleeding from the right wall is not present in Fig. 6b and is slightly present in Fig. 6c. Moreover, a larger radius for the sample pattern spreads the samples too much that most of them don't reside in screen space. This is the reason why the green color bleeding of the Lucy right wing has a low intensity. However, because the radius is larger, there are more potential sources of information, therefore all the indirect diffuse lighting present in Fig. 6b is also present in Fig. 6c with a higher intensity. It is possible to use an intensity scale factor to obtain similar intensities between figures, but all the intensities that are already have a small influence, like the green color bleeding on the right wing, will be virtually erased. Our use of the RSM to acquire the indirect diffuse lighting for large distances prevented those problems. Moreover, we took advantage of the good visual results provided by the SSDO technique for short distances (Fig. 6b).

The original RSM technique, on the other hand, could provide good results for long distances, but lacks the ability to provide stable results for short distances.

a) Original RSM (R=200)

b) Original SSDO (S=200, r=3)

c) Original SSDO (S=200, r=30)

d) Hybrid (R=200, S=200)

Fig. 6. Indirect diffuse lighting and ambient occlusion comparison of our hybrid global illumination approach (d) and the original Reflective Shadow Mapping (a) and Screen Space Directional Occlusion (b, c) techniques. The SSDO technique is presented with two different radii for the sampling pattern (R is the number of samples for the RSM and S is the number of samples for the SSDO).

The local 2D space of the sample pattern does not allow the control of the samples position in world space, therefore the Lucy head and left wing become over-illuminated in Fig. 6a. Also, at the bottom of the image, the clothing receives blue color bleeding from the floor, which is not physically correct. Moreover, the 2D space of the sample pattern i ntroduces a visual problem known as banding artifact. This phenomenon is present in the noise pattern of the right wing in Fig. 6a. Our use of the local 3D space for the sample pattern prevented this artifact to appear. Furthermore, we replaced the RSM technique for short distances with the more suited SSDO. In this way, we achieved a stable version of both techniques that provides the best possible visual results.

6. Conclusions

We presented a novel hybrid global illumination approach that uses information from multiple framebuffers acquired from the observer position and light positions. We presented a sampling strategy to acquire the indirect diffuse lighting with those framebuffers and provided an approach to combine the indirect specular lighting and ambient occlusion maps. We also presented several extensions to accommodate different light types, to produce subsurface scattering and the direct lighting for area lights. We presented different optimization strategies to scale the approach for multiple lights.

We proved in this paper that our method could provide real-time performances and good visual results. Also, we demonstrated that it is a stable global illumination method that eliminates the visual problems of the original techniques which it relies on. To avoid the overlapping of samples between the RSM and the SSDO, we changed the 2D sample pattern of the RSM with a 3D one, thus solving the visual artifacts of the original RSM technique. Also, by using information obtained from different points of view, we avoided the disadvantages of the screen space techniques. Moreover, our method could take advantage of any optimization approach for the Reflective Shadow Mapping and Screen Space Directional Occlusion techniques to enhance the visual results and the performance.

Acknowledgements

The work has been funded by the Operational Programme Human Capital of the Ministry of European Funds through the Financial Agreement 51675/09.07.2019, SMIS code 125125.

R E F E R E N C E S

[1] A. Appel, “Some techniques for shading machine renderings of solids,” in Proceedings of the April 30–May 2, 1968, spring joint computer conference, 1968, pp. 37–45.

[2] J. T. Kajiya, “The rendering equation,” in Proceedings of the 13th annual conference on Computer graphics and interactive techniques, 1986, pp. 143–150.

[3] M. F. Cohen and J. R. Wallace, Radiosity and realistic image synthesis. Elsevier, 2012.

[4] H. W. Jensen, “Global illumination using photon maps,” in Eurographics workshop on Rendering techniques, 1996, pp. 21–30.

[5] A. Keller, “Instant radiosity,” in Proceedings of the 24th annual conference on Computer graphics and interactive techniques, 1997, pp. 49–56.

[6] G. J. Ward, F. M. Rubinstein, and R. D. Clear, “A ray tracing solution for diffuse interreflection,” in Proceedings of the 15th annual conference on Computer graphics and interactive techniques, 1988, pp. 85–92.

[7] P. H. Christensen and D. Batali, “An Irradiance Atlas for Global Illumination in Complex Production Scenes.,” in Rendering Techniques, 2004, pp. 133–142.

[8] P.-P. Sloan, J. Kautz, and J. Snyder, “Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments,” in Proceedings of the 29th annual conference on Computer graphics and interactive techniques, 2002, pp. 527–536.

[9] C. Dachsbacher and M. Stamminger, “Reflective shadow maps,” in Proceedings of the 2005 symposium on Interactive 3D graphics and games, 2005, pp. 203–231.

[10] S. Laine, H. Saransaari, J. Kontkanen, J. Lehtinen, and T. Aila, “Incremental Instant Radiosity for Real-Time Indirect Illumination.,” in Rendering Techniques, 2007, pp. 277–286.

[11] R. Prutkin, A. Kaplanyan, and C. Dachsbacher, “Reflective Shadow Map Clustering for Real-Time Global Illumination.,” in Eurographics (Short Papers), 2012, pp. 9–12.
[12] G. Nichols, J. Shopf, and C. Wyman, “Hierarchical image-space radiosity for interactive global illumination,” in Computer Graphics Forum, 2009, vol. 28, no. 4, pp. 1141–1149.
[13] A. Kaplanyan and C. Dachsbacher, “Cascaded light propagation volumes for real-time indirect illumination,” in Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games, 2010, pp. 99–107.
[14] S. Thiedemann, N. Henrich, T. Grosch, and S. Müller, “Voxel-based global illumination,” in Symposium on Interactive 3D Graphics and Games, 2011, pp. 103–110.
[15] C. Crassin, F. Neyret, M. Sainz, S. Green, and E. Eisemann, “Interactive indirect illumination using voxel cone tracing,” in Computer Graphics Forum, 2011, vol. 30, no. 7, pp. 1921–1930.
[16] P.-P. Sloan, N. K. Govindaraju, D. Nowrouzezahrai, and J. Snyder, “Image-based proxy accumulation for real-time soft global illumination,” in 15th Pacific Conference on Computer Graphics and Applications (PG’07), 2007, pp. 97–105.
[17] G. Nichols, J. Shopf, and C. Wyman, “Hierarchical image-space radiosity for interactive global illumination,” in Computer Graphics Forum, 2009, vol. 28, no. 4, pp. 1141–1149.
[18] C. Soler, O. Hoel, and F. Rochet, “A deferred shading pipeline for real-time indirect illumination,” in ACM SIGGRAPH 2010 Talks, 2010, p. 1.
[19] T. Ritschel, T. Grosch, and H.-P. Seidel, “Approximating dynamic global illumination in image space,” in Proceedings of the 2009 symposium on Interactive 3D graphics and games, 2009, pp. 75–82.
[20] T. Sousa, N. Kasyan, and N. Schulz, “Secrets of CryENGINE 3 graphics technology,” 2011.
[21] M. Deering, S. Winner, B. Schediwy, C. Duffy, and N. Hunt, “The triangle processor and normal vector shader: a VLSI system for high performance graphics,” ACM Siggraph computer graphics, vol. 22, no. 4, pp. 21–30, 1988.
[22] G. Miller, “Efficient algorithms for local and global accessibility shading,” in Proceedings of the 21st annual conference on Computer graphics and interactive techniques, 1994, pp. 319–326.
[23] À. Méndez-Feliu and M. Sbert, “From obscurances to ambient occlusion: A survey,” The Visual Computer, vol. 25, no. 2, pp. 181–196, 2009.
[24] M. Mittring, “Finding next gen: Cryengine 2,” in ACM SIGGRAPH 2007 courses, 2007, pp. 97–121.
[25] B. T. Phong, “Illumination for computer generated pictures,” Communications of the ACM, vol. 18, no. 6, pp. 311–317, 1975.
[26] K. Vardis, G. Papaioannou, and A. Gaitatzes, “Multi-view ambient occlusion with importance sampling,” in Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, 2013, pp. 111–118.
[27] C. Dachsbacher and M. Stamminger, “Translucent shadow maps,” Rendering Techniques, vol. 2003, pp. 197–201, 2003.
[28] J. Jimenez, V. Sundstedt, and D. Gutierrez, “Screen-space perceptual rendering of human skin,” ACM Transactions on Applied Perception (TAP), vol. 6, no. 4, pp. 1–15, 2009.
[29] O. Nalbach, T. Ritschel, and H.-P. Seidel, “Deep screen space,” in Proceedings of the 18th meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, 2014, pp. 79–86.